Summer 2007

CEG 460/660-01: Introduction to Software Computer Engineering

Eric Maston
Wright State University - Main Campus

CEG460/660 Introduction to Software Engineering

Summer Quarter 2007

Wright State University

Course Description
This course is concerned with the techniques of designing and constructing large programs. Some of the required basic concepts necessarily have to be developed using small programs as examples. To this extent, we also study programming-in-the-small. The overall objectives are to present an overview of issues in the development of software, to discuss terminology, to illustrate via example case studies, and to give sufficiently detailed advice on how to develop quality software. Hands-on experience is emphasized through the use of homework and a class project.

Professor
Eric Matson
Office: 336 Russ Engineering Center
Office Hours: T/U 3:00 – 4:10 or by appointment or email.
Office Phone: 775-5108
Email: eric.matson@wright.edu
Class: Tuesday/Thursday 4:10-5:35, Russ Engineering Center 153

Text
Bernd Bruegge and Allen H. Dutoit, *Object-Oriented Software Engineering: Using UML, Patterns, and Java, 2nd Edition,* Prentice Hall, 2004.

Prerequisites
CS400 or CS600

Grading
Grading will be as follows:

Homework	15%
Project	25%
Midterm	30%
Final Exam	30%

Course grades will be based on the total score as follows. A: 90-100, B: 80-89, C: 70-79, D: 60-69, F: below 60. Grades may be further curved if appropriate.

Project
There will be a term project in class in which you will develop a significant application. The project will be worked in teams of three. You may pick your partners or I will pick them. More detail on the project will be handed out later.

Policies and Notes

- Attendance: Attendance is not required, nor will it be taken after the first couple of lectures. If you are not a regular attendee, it will be your responsibility to seek out what material was covered in the lecture and learn it. Most of my exam questions will be taken directly from ideas covered during the lecture, so it greatly helps if you attend!

- I will utilize WebCT to post updates to the course, sample code, projects, announcements, schedule, etc. Get in the habit of checking it regularly.

- Always make back ups of all of you work. Never have just one copy of anything!

- If you are going to miss an exam or be late on any submitted material, for any reason, discuss it with me in advance. If it is an emergency situation, please notify me as soon as possible.

- You can reach me a number of ways. Email is normally the best as I check it about 18 hours a day normally. You can also reach me by phone during the day at 775-5108. If you need human contact either stop in during my office hours, make an appointment, or just come by my office. If I am in and not on a deadline to get something else completed, I will normally try to help as much as possible.

- There are technologies we will use in this class that you may not already know, such as file transfer, command line, text editors, file systems, etc. You are responsible for learning these technologies, but we will discuss them in class as necessary.

Academic Misconduct

In this class, the only way to truly learn the concepts to is do the work yourself. I encourage working with other people on the course concepts. When you begin to write the program, complete and submit your own work.

Work that has obviously been copied or in the more extreme case, when the original author's name has not even been changed, both parties will receive a 0 grade for that assignment. Both parties will also be turned over to the Office of Judicial Affairs.

Schedule (always subject to changes)

#	Day	Date	Topic	Reading
1	T	June 12	Introduction, Software Lifecycles	Chapter 1
2	U	June 14	Lifecycles and Process	15 (skip 15.2)
3	T	June 19	Requirements,	2, 4
4	U	June 21	Ethics, Project	Handouts
5	T	June 26	UML, Analysis	2, 5, Handouts
6	U	June 28	Analysis	
7	T	July 3	Object Design	
8	U	July 5	Object Design	8
9	T	July 10	Object Design	9,12
10	U	July 12	Object Design, Review	
11	T	July 17	**Midterm Exam**	
12	U	July 19	System Design	6, 7
13	T	July 24	Implementation	10, Handouts
14	U	July 26	Testing	11
15	T	July 31	Testing	
16	U	Aug 2	Testing	
17	T	Aug 7	Structured Analysis	
18	U	Aug 9	Structured Design	
19	T	Aug 14	Maintenance, Review	
20	U	Aug 16	**Final Exam**	

Always have readings scheduled for that day complete prior to the class meeting

Note: T = Tuesday
U = Thursday